Amendments to the Specifications:

Please replace paragraph **[0014]** with the following amended paragraph:

[0014] Liang et al. discloses a patch-based sampling algorithm for synthesizing textures from a sample. This sampling algorithm is hereafter explained in reference to Fig. 1A where an image 2 has a target area 4 to be filled with textures from a sample texture 6. Target area 4 is divided into target patches B_k having a patch size of w by w (only one is labeled for clarity), where k is a variable. If the width or the height of target area 4 is not a multiple of w, the size of the last row and the last column of target patches is defined as:

$$w * w_1,\ w_2 * w, \tag{1}$$

$$w_1 = \mathrm{W}\ mod\ w\ ,\ w_2 = \mathrm{H}\ mod\ w, \tag{2}$$

where W and H express the width and height <u>of the target area</u>, respectively, and *mod* is the function that calculates the residual of W or H divided by w.

Please replace paragraph **[0017]** with the following amended paragraph:

[0017] The corresponding points in boundary zones $E_{B_{(x,y)}}$ and E_{B_k} are compared to determine if a sample patch $B_{(x,y)}$ matches a target patch B_k. If the distance (i.e., the difference) between the sample patch $B_{(x,y)}$ and the target patch B_k is less than a prescribed threshold, then that sample patch $B_{(x,y)}$ is placed in a set ψ_B. The definition of set ψ_B is:

$$\cancel{\Psi_B = \{B_{(x,y)} \mid d(E_{B_{(x,y)}}, E_{B_k}) < d_{max}\}}\ \underline{\Psi_B = \{B_{(x,y)} \mid d(E_{B_k}, E_{B_{(x,y)}}) < d_{\max}\}}, \tag{3}$$

where d is the distance between boundary zones $E_{B_{(x,y)}}$ and E_{B_k} of sample patch $B_{(x,y)}$ and target patch B_k, respectively, and d_{max} is the prescribed threshold. The definition of distance d between boundary zones $\cancel{E_{B_K}}$ $\underline{E_{B_k}}$ and $E_{B_{(x,y)}}$ is:

$$d(E_{B_k}, E_{B_{(x,y)}}) = [\frac{1}{A}\sum_{i=1}^{A}(p^i_{B_k} - p^i_{B_{(x,y)}})^2]^{1/2}, \quad (4)$$

where A is the number of corresponding points in boundary zones $E_{B_{(x,y)}}$ and $\cancel{E_{\overline{B_K}}}$ $\underline{E_{B_k}}$, and $p^i{}_{B_k}$ and $p^i{}_{B_{(x,y)}}$ denote the corresponding gray values of the corresponding points.

Please replace paragraph **[0019]** with the following amended paragraph:

[0019] After filling in one target patch, the texture of that target patch becomes part of the known boundary zones of adjacent target patches. For example in Fig. 1B, a ~~left~~ right area of a boundary zone $\cancel{E_{\overline{B_{K+1}}}}$ $\underline{E_{B_{k+1}}}$ from a target patch B_{k+1} becomes a known ~~right~~ left area of a boundary zone $\cancel{E_{\overline{B_{K+2}}}}$ $\underline{E_{B_{k+2}}}$ from a target patch B_{k+2}. After sample patches fill in the target patches, the overlapping areas of their boundary zones are blended.

Please replace paragraph **[0023]** with the following amended paragraph:

[0023] Fig. 3 illustrates a method 10 for filling a target area 102 (Fig. 4) in an image 104 (Fig. 4) with textures from a sample ~~texture~~ area 106 (Fig. 4) in one embodiment of the invention. Target area 102 may be a damaged area that needs to be repaired while sample area 106 may be any undamaged area outside of target area 102 in image 104. Alternatively, sample area 106 can be another image or a group of sample patches. Method 10 can be implemented with software on a computer or any equivalents thereof.

Please replace paragraph **[0026]** with the following amended paragraph:

[0026] In step 16, the computer divides target area 102 into target patches B_k with associated boundary zones $\cancel{E_{\overline{B_K}}}$ $\underline{E_{B_k}}$.

Please replace paragraph **[0030]** with the following amended paragraph:

[0030] In steps 24 and 26, the computer determines the distance between the current target patch B_k and the current sample patch $B_{(x,y)}$. Specifically, in step 24, the computer determines the difference

between the corresponding points in boundary zones $\cancel{E_{B_K}}$ $\underline{E_{B_k}}$ and $E_{B_{(x,y)}}$. Unlike Liang et al., the computer divides the boundary zones into boundary areas and then determine the differences between the corresponding boundary areas as follows:

$$d_n = \left[\frac{1}{A_n}\sum_{i=1}^{A_n}(p^i_{B_k} - p^i_{B_{(x,y)}})^2\right]^{1/2}, \ 2 \le n \le 4, \tag{6}$$

where d_n is the difference of the nth pair of corresponding boundary areas in boundary zones $\cancel{E_{B_K}}$ $\underline{E_{B_k}}$ and $E_{B_{(x,y)}}$, A_n is the number of corresponding points in the nth pair of corresponding boundary areas, and $p^i{}_{B_k}$ and $p^i{}_{B_{(x,y)}}$ denote the corresponding gray values of the corresponding points. In one embodiment, the computer divides each boundary zone into a top boundary area B_{top}, a left boudnary area B_{left}, a bottom boundary area B_{bot}, and a right boundary area B_{right} as shown in Fig. 5. Note that the four corners are part of two boundary areas and will be calculated twice in equation 6.